Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 61 (2013) 227 – 232

Procedia Engineering

www.elsevier.com/locate/procedia

Parallel Computational Fluid Dynamics Conference (ParCFD2013)

Parallel Large Eddy Simulations of Wind Farms with the Actuator Line Method

A. Baez-Vidal[a,*], O. Lehmkuhl[b], D.M. Valdivieso[a], C.D. Pérez Segarra[a]

[a]*Centre Tecnològic de Transferència de Calor, ETSEIAT c\Colom 11, 08222, Terrassa Barcelona, Spain*
[b]*TermoFluids S.L. Avinguda Jaquard, 97 1-E, 08222, Terrassa, Barcelona, Spain*

Abstract

Parallel Computation of wind farm Large Eddy Simulations (LES) requires the use of with Wind Turbine Models (WTM). CFD and WTM demand different domain decompositions as optimal CFD and WTM decompositions do not necessarily coincide. Nevertheless, data exchange between CFD and WTM must not penalize overall simulation performance. A coupling strategy for data exchange is described and has been tested. It enables the parallel simulation of wind farms with WTM. Simulations of wind turbine wakes have been achieved. Preliminary results show the parallelization works properly but that the simulations do not resolve flows with enough accuracy.

Selection and peer-review under responsibility of the Hunan University and National Supercomputing Center in Changsha (NSCC)
Keywords: Wind Energy, CFD, actuator line, LES

Nomenclature

R	radius of wind turbine rotor (m)
O	Owned domain for a CPU ($cell$)
H	Halo domain for a CPU ($cell$)
Dw	domain of a wind turbine ($cell$)
Da	domain of an airfoil ($cell$)
ca	cell occupied by an airfoil ($cell$)
$\mathbf{u}$	Velocity (m/s)
$\mathbf{n}$	Unitary vector of $\Re^3$ normal to a surface ($\cdot$)
$\mathbf{e}$	Unitary vector of $\Re^3$ ($\cdot$)
p	pressure (Pa)
c_L	lift coefficient ($\cdot$)
c_D	drag coefficient ($\cdot$)
c	arifoil chord (m)

Greek symbols

ρ	density (kg/m^3)
Ω	finite volume (m^3)
$\partial\Omega$	surface enclosing a volume (m^2)
δ	infinitesimal ($\cdot$)
σ	finite surface (m^2)
$\underline{\underline{\tau}}$	Viscous stresses tensor (m^2)

Subscripts and superscripts

i, j	CPU identifier, coordinate
k	Wind Turbine identifier

* Corresponding author. Tel: (+34) 93 739 81 92 Fax: (+34) 93 739 89 20
E-mail address: aleix@cttc.upc.edu

1877-7058 © 2013 The Authors. Published by Elsevier Ltd. Open access under CC BY-NC-ND license.
Selection and peer-review under responsibility of the Hunan University and National Supercomputing Center in Changsha (NSCC)
doi:10.1016/j.proeng.2013.08.008

1. Introduction

Wind energy production is moving towards bigger wind turbines composing bigger wind farms. Offshore wind farms can take advantage of the most favorable wind. New complex structural challenges for wind turbines arise as rotor radii increase and floating platform dynamics interact with rotors'. Unsteady structural analysis become relevant in order to ensure operational safety and to reduce maintenance costs. In this context, for the airflow predictions, LES takes advantage with respect to Reynolds Averaged Navier Stokes (RANS) simulations, that calculate only average magnitudes. LES enables the calculation of the unsteady fluid fields, thus a deeper knowledge about fatigue loads, extreme wind situations, platform-rotor dynamics couplings and wakes. Moreover, for onshore wind farms, LES can predict complex terrain unsteady effects on air streams can be of high impact on project feasibility while RANS can not.

The spatial and temporal domains within LES simulations should run can become of the order of $10x10x0.5km$. The volume occupied by a wind turbine and it's nearest wake can be estimated to be $3Rx2Rx0.5R = 3R^3$, i.e. $6.5 \cdot 10^5 m^3$ for $R = 60m$. Taking into account that Reynolds numbers for such flows is determined by blade chord $c \sim 1m$, atmospheric air density and flow velocity relative to blades, this leads to $Re \simeq 10^7$, giving a Kolmogorov length of $\eta = 5.62 \cdot 10^{-6}m$. Resolving a LES with the smallest scale 100η, would lead to $N = 6.5 \cdot 10^5/(5 \cdot 10^{(-4)})^3 = 5 \cdot 10^{15}$ control volumes. It is clearly unfordable in terms of computational cost. Even sub-resolving the flow in the most of the domain around a turbine, the number of control volumes needed would be enormous. Hence, if the domain containing a whole wind farm is to be simulated, wind turbine models must be considered.

Two WTM have been extensively used in the literature: the actuator disk method, and the actuator line method. The actuator disk concept is used by Jiménez and Crespo in combination with a channel flow LES solver [1] , [2] can predict statistical wake properties but the model used does not capture rotating dynamics and frequencies. Another weak point for the Actuator disk is the disregard of the tower effect, which changes the wake behavior in a significantly. The actuator line was developed by Sørensen and Shen. It allows the introduction of the unsteady effects and, if wanted tower and nacelle effects.

Clustered wind turbines turbines fatigue loading and power production calculations require complex models for wakes [3]. It is not fully known how these wakes behave mainly due to measurements uncertainties in wind farm environments. The detail that LES with AL can offer about turbulent wakes is necessary to generate better wind farm models. In this context, the National Renewable Energy Laboratory has implemented the actuator line method with an OpenFOAM CFD solver while Sørensen, Troldoborg and others have simulated wind turbines and groups of wind turbines with the same technique and EllipSys3D [4]. In both studies the momentum equations were resolved with PISO solvers. A parallelization strategy for the coupling between the explicit Fractional Step Method and Actuator Line is described in this work.

2. The Actuator Line Method on a Finte Volumes CFD simulation

2.1. Actuator Line Method

The actuator line method consists on modeling forces produced by an aerodynamic body as a set of body force singularities. For a Finite Volumes discretization, these body forces depend stiffly on the velocities at the cell they are applied. For wind turbines, blades are modeled as a set of aligned 2D airfoils whose position varies on time describing their trajectory. Lift and drag depending on local instantaneous velocity of the mentioned airfoils should be applied at the corresponding cells. Note that various aerodynamic sections could coincide in one control volume and that these sections could belong to different wind turbine blades.

The Navier Stokes equations (NSE) are, for a control volume Ω:

$$\int_{\partial\Omega} \rho u_i n_i \delta\sigma = 0$$

$$\frac{\partial \int_\Omega \rho u_i \delta\Omega}{\partial t} + \int_{\partial\Omega} \rho u_i u_j n_j \delta\sigma = -\int_{\partial\Omega} \partial_i p \, \delta\sigma + \int_{\partial\Omega} \tau_{ij} n_j \delta\sigma + \int_\Omega (\rho b_i + f_i)\delta\Omega \tag{1}$$

The actuator line model stablishes:

$$f_i = \frac{1}{2}\rho u^2 \mathbf{e_i} \cdot \sum_j \left[c(r)\delta r\,(c_L \mathbf{e_L} + c_D \mathbf{e_D})\right]^j$$

The actuator line is a model for the computations of body forces $f_i + b_i$ within a Navier-Stokes solver. In our work, the chosen discretization of the Navier-Stokes equation is the kinetic energy preserving [5] because of it's good non-dissipative properties [6]. The chosen LES model is WALE [7] as it is suitable for unstructured meshes and has performed optimally for flows without laminar to turbulent transitions within limit layers [8].

For the calculation of the force done by a section, velocities are first passed to the rotor frame of reference. Next, aerodynamic forces are computed from airfoil data. All forces generated by a rotor are responsible of it's torque and power generation. As rotors turn, the positions affected by the aerodynamic forces vary according to rotor mechanics and must be recomputed. From the method itself yields the necessity to communicate all forces originated by a rotor in order to compute it's power output. The algorithm describing the actuator line method is represented in figure 1.

2.2. Algorithm Parallelization

If several CPU's used, the WTM and the CFD solver require different decompositions on the same geometrical discretization. The reason is that for the CFD domain decompositions are established in order to reduce communication, that is,to minimize halos. For the Actuator Line Method, at a given instant of simulation, each turbine defined should receive data from the CFD simulation at the cells occupied by it's airfoils. It is intended to make the CFD decompositions as much independent as possible and not be constrained by a particular wind farm configuration that might ensure that all cells from which each turbine would require data were attributed to the same CPU.
The cell within the point that an airfoil j of a wind turbine i lays, for a trajectory $C = \{\mathbf{x}(\psi), \psi \in [\psi_1, \psi_2] \subseteq \Re\}$ for $\psi = \psi^*$ is $ca_{ij}(\psi^*)$. Then, $Da_{ij} = \cup_C ca_{ij}(\psi)$ it the set of all the cells that an airfoil j of a turbine i would intersect. For the wind turbine i,$Dw_i = \cup_j Da_{ij}$ represents all the cells that one of it's airfoils would intersect while $Cw_i(\psi^*) = \cup_j ca_{ij}(\psi^*)$. If the trajectories of all airfoils is known a priori $Dai\,j$ and Dw_i can be precomputed. Finally, for rotors with only one degree of freedom (i.e. only rotation around rotor axis allowed) or for any other closed trajectory the sequence of cells intersected satisfies that, $ca_{ij}(\psi^*) = ca_{ij}(\psi^* + 2\pi/T)$. Figure 1 shows the domain of a wind turbine on three CFD domain decompositions.

Within the CFD simulation and depending on CPU domain decompositions, cells are attributed 2 different numberings : The Local Numbering (LN) , and the Global Numbering (GN). Reading and writing data by the local numbering is much faster than by the global. The reason is that the local numbering corresponds to the position in a local container whereas the global numbering is not, so reading and writing data by the global numbering requires the translation of the global position to be into local numbering, i.e. a search (normally the search of an integer in a sorted list). Cell's GN is unique for all the domain and allows the unequivocal identification of each cell, which is necessarily used when data is to be transferred to functions not sharing domain decomposition with CFD. In this work the CFD global numbering has been used for that purpose.
Within the WTM solver it is needed that for each wind turbine k one CPU acts as master M_k while CPUs with $O_j \cap C_k(\psi^*) \neq \emptyset$ act as slaves Sl_k^j. The cause for this is that some of the most simulation's relevant results are wind turbine performances, so all aerodynamic forces done by the WTM should be, in some point of execution and for every iteration, gathered in M_k. After this statement, the method chosen has been that Sl_k^j read velocities and send to M_k, which computes all forces and sends to Sl_k^j to be inserted into the CFD simulation. Other methods would better distribute the computational load when computing $\mathbf{f}$ but would require a perfect synchronization of the calculation of

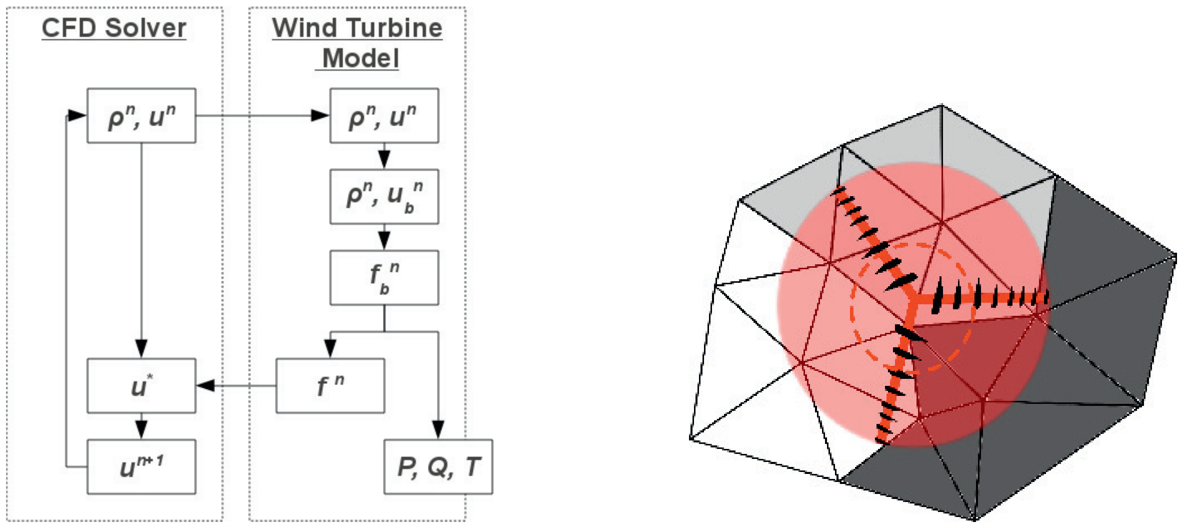

Fig. 1. Left: Algorithm for a CFD simulation including simplified wind turbine models. Right: Domain of a wind turbine (translucent red) on the CFD decomposition (grey intensities). Airfoils (black) are distributed along blades (opaque red)

$C_k(\psi)$, that changes on time according to the wind turbine dynamics. It was found that if the geometrical tolerance used to precompute the airfoil's trajectories was not machine precision and rotor torque equilibrium was used for the calculation of the rotational speed, not all CPUs always computed the same $C_k(\psi)$ and simulations failed. For robustness reasons this approach was dismissed. More strategies can be envisaged.
In this document, M_k is chosen as the CPU with a biggest $O_i \cap D_k$.

Then the parallel algorithm reads, for the i_{th} CPU and k_{th} Wind Turbine and with $\mathbf{u}^n(O_i \cup H_i)$ known by i:

1. WTM(only master): Compute $C_k(\psi^*)$
2. WTM: Read $\mathbf{u}(Dw_k \cap O_i)$ and communicate to master
3. WTM (only master) :Compute $\mathbf{f}(C_k(\psi^*))$ and communicate to slaves
4. CFD: Use $\mathbf{f}(Dw \cap O_i)$ to compute $\mathbf{u}^{n+1}(O_i)$
5. CFD: Communicate $\mathbf{u}^{n+1}(H_i)$
6. go to 1.

In order to execute this algorithm efficiently, the WTM is provided with 2 different domain decompositions: one that facilitates data reading and the other that facilitates data writing. They are depicted in figure 2.

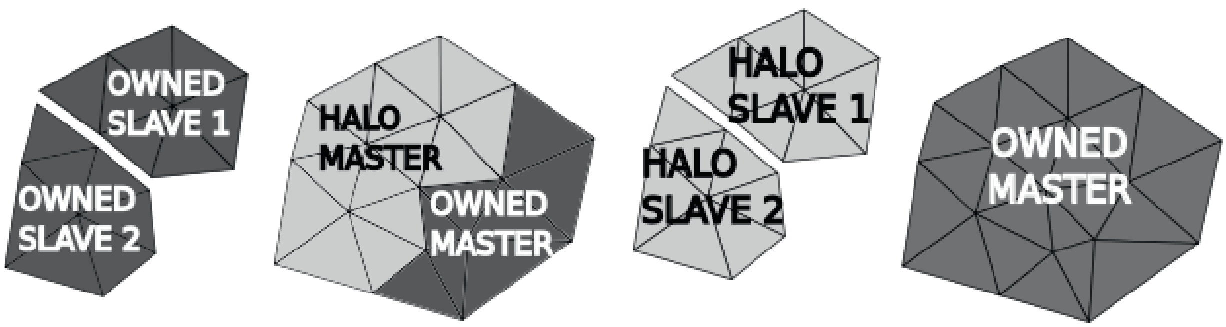

Fig. 2. Left: Velocity reading domain. Right: Force writing domain.

3. Preliminary results

Preliminary results show the implemented method for the treatment of the coupling between enable the resolution of wind farms LES with the Actuator Line wind turbine models for any mesh, structured or unstructured. Validation of the method is to be carried out by comparing simulations of the MEXICO [9] project with experimental data for wake properties. For power, toque and thrust output the benchmark case is the NREL phase VI [10] unsteady aerodynamics experiment conducted at the AMES research center.

The first numerical experiment carried out is the reproduction of the NREL phase VI UAE [10] the used mesh is generated by extruding a 2D mesh of 2160 octahedra in 32 planes, the sizes of cells being $0.6x0.5x1.2m$ at the highest density zone (i.e. near the turbine position). Set up for the turbine is pitch angle 3 and wind velocity $15m/s$. Tower and nacelle were included by means of the immersed boundary method. First results show that the method implemented produces high amplitude thrust output oscillations as can be seen in figure 3. I is also observed that the thrust is underestimated wile power is overestimated. The huge oscillation on results is probably due to tower effects, as 2 thrust minimums are observed for each rotor cycle, as it corresponds to the tower effect to a 2 blades turbine as is the case. Computations were carried out with a very coarse meshes that could not resolve well pressure near the immersed boundary. No filter on the actuator line forces was applied as in other authors works [11]. Aerodynamic coefficients where extracted from [12].

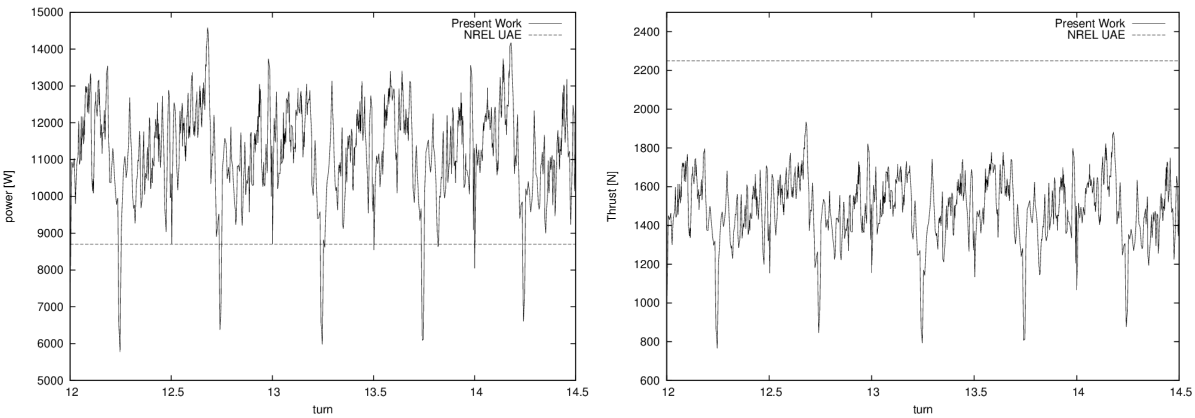

Fig. 3. Left: Power output for the NREL phase VI experiment. Computed power output overestimates the measured. Right: Thrust output for the NREL phase VI experiment. Computed thrust underestimates the measured.

Another simulation carried out was the wake of a Offwindtech wind turbine, which diameter is 120m. It was done in order to test the LES-ALM in a massive parallel computation. Result of velocity magnitude of a section parallel to the wind direction is shown in figure 4.

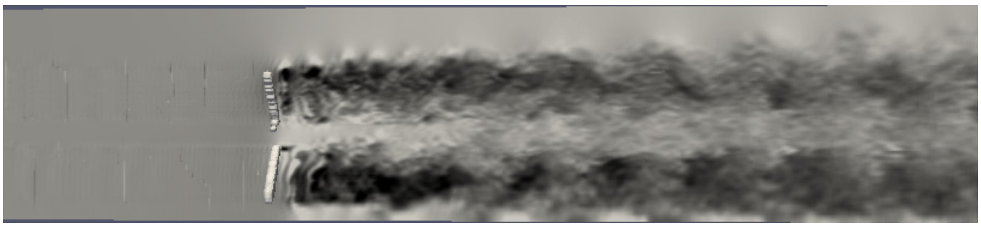

Fig. 4. Offwindtech wind turbine velocity magnitude.

4. Further Work

The parallelization of the LES-ALM method has been achieved for any mesh. However, first simulation results show that, although in the same order of magnitude as benchmark experiments, the accuracy of the numerical method is not acceptable. A very strong dependency of results to the input airfoil characteristics and to the filter employed for the forces described by other ALM authors has been observed. Hence, the first objective is now to resolve inaccuracies and to establish what force filters are to be used in order to bring the LES-ALM to it's maximum capabilities with complex terrain and tower and nacelle effects. In this sense, it is expected that appropriate filters can avoid the oscillations amplitude and provide accurate power and thrust outputs. After that, wind farms flows will be studied.

Acknowledgements

This work has been granted by the spanish Secretará de Estado de Investigación, Desarrollo y Competitividad of the Ministerio de Economía y Competitividad BES-2010-032414 grant for the ENE2009-07689 project.

References

[1] A. Jimenez, A. Crespo, E. Migoya, and J. Garcia. Advances in Large-Eddy Simulation of a wind turbine wake. In *The Science of Making Torque from Wind*, pages 1–14, 2007.

[2] A. Jimenez, A. Crespo, and E. Migoya. Application of a LES technique to characterize the wake defl ection of a wind turbine in yaw. *Wind Energy*, 13(0):559–572, 2010.

[3] S. Frandsen. *Turbulence and turbulence generated structural loading in wind turbine clusters*. Riso National Laboratory, 2007.

[4] N. Troldborg, G. Larsen, H. Madsen, K. Hansen, N. Sorensen, and R. Mikkelsen. Numerical simulations of wake interaction between two wind turbines at various inflow conditions. *Wind Energy*, 14:859–876, 2011.

[5] R. W. C. P. Verstappen and A. E. P. Veldman. Symmetry-preserving discretization of turbulent flow. *Journal of Computational Physics*, 187(1):343–368, 2003.

[6] I. Rodriguez, R. Borrell, O. Lehmkuhl, C. D. Perez-Segarra, and A. Oliva. Direct numerical simulation of the flow over a sphere at $Re = 3700$. *Journal of Fluid Mechanics*, 25(25):263–283, 2011.

[7] F. Nicoud and F. Ducros. Subgrid-scale stress modeling based on the square of the velocity gradient tensor. *Flow, Turbulence and Combustion*, 62:183–200, 1999.

[8] O. Lehmkuhl, A. Baez, I. Rodrguez, and C. D. Prez Segarra. Direct numerical simulation and Large-Eddy simulations of the turbulent flow around a NACA-0012 airfoil. In *Proceedings of the 7th International Conference on Computational Heat and Mass Transfer*, pages 1–8, 2011.

[9] JG. Schepers, T. Cho, S. Gomez-Iradi, P. Schaffarczyk, A. Jeromin, W. Z. Shen, T. Lutz, K. Meister, B. Stoevesandt, S. Schreck, D. Micallef, R. Pereira, T. Sant, HA. Madsen, and N. Sorensen. Analysis of Mexico wind tunnel measurements. Technical report, Energy research Centre of the Netherlands, 2012.

[10] S. Schreck. The NREL Full-Scale Wind Tunnel Experiment. *Wind Energy*, 5(5):77–84, 2003.

[11] N. Troldborg. *Actuator Line Modeling of Wind Turbine Wakes*. PhD thesis, Technical University of Denmark, 2008.

[12] M. M. Hand, D. A. Simms, L. J. Fingersh, D. W. Jager, J. R. Cotrell, S. Schreck, and S. M. Larwood. Unsteady Aerodynamics Experiment Phase VI: Wind Tunnel Test Configurations and Available Data Campaigns. Technical report, National Renewable Energy Laboratory, 2001.